SFWR ENG 3GB3: A New Game

Due on Part 1: Friday October 10th. Part 2: Friday October 24th.

Dr. Jacques Carette

Idea

You are to design and implement a game of your choice. You are given a small set of requirements and your task is to create a game that meets the requirements. The assignment is divided into two parts.

Part 1 - Proposal

The first part is the design of the game. You are to write a detailed proposal of the game you intend to implement.

The proposal should include the following:

- A detailed description of the game - This includes giving details of the game rules, how it will work, any features that the game will have, etc.
- A description of how each requirement will be met in your proposed game

You will submit your proposal for review. The TA and the professor will review your proposal for degree of difficulty. Remember that you will need to implement the proposed game in a relatively short period of time, so do not make it too difficult, but also do not make it too simple. Use your judgement to design a game that will have a feasible implementation.

**NOTE: If you are having difficulty coming up with a creative game or are unsure of the feasibility of an idea, please contact the TA or the professor and they will assist you.

Part 2 - Implementation

Physics Requirements

- The game must have a rotational aspect along all axes (x, y, and z)

Rendering

- The game must be rendered in 3D

Core

- The game must take user user input from the keyboard or mouse
- The game must demonstrate an appropriate modular design
- The game must be a game; i.e. there should be some playable aspect allowing a player to win or lose

Submission Requirements

- README file containing notes on what libraries you used and how to get them (if not included)
- Reference(s) file containing notes on what references you used. This includes books, web pages, discussion with friends, etc. This is an 'engineering journal'. [This can be appended to README]
- A short User Manual so that someone can learn to play your game. This should contain controls, rules, etc.
- Makefile or build script or detailed instructions on how to compile
- Source code
- Sample user input for you game
- Must be submitted as a single file (zipped or gziped archive)

Notes

- Must run on Linux, Mac OSX, Windows XP or Windows Vista using OpenGL
- Each student does own assignment (no group work this time)
- You may 'borrow' source code from the web for use in your game as long as it is properly referenced.
- Any programming language can be used
- Pre-written linear algebra and ordinary differential equation integration libraries are recommended, particularly GSL (http://www.gnu.org/software/gsl/)

Marking

- 20% proposal
- 70% implementation
  - 30% rotation
  - 15% rendering
  - 15% playability
  - 10% modularity
- 10% software engineering (documentation, etc.)

Bonus

- Bonus marks will be awarded on a case-by-case basis. Your game will be judged according to your proposal as a baseline, and features (including software engineering qualities) that go beyond the baseline will be worth a certain amount of bonus marks.